

요약

- 본 논문은 변종 악성코드 탐지를 목적으로 CNN 기법을 이용하여 악성코드 탐지율을 높이고 오탐률은 낮추도록 구현
- CNN 기법의 효율을 검증하기 위해 기존 악성코드 탐지 기법과 비교 분석
- 악성코드의 이미지화를 통해 변종 악성코드를 분류 및 탐지

서론

- 변종 악성코드가 개발되어 제로데이 [1], 랜섬웨어 [2]와 같은 다양한 공격으로 인해 사이버 피해가 늘고 있음 [3]
- 악성코드를 탐지하기 위해 많은 악성코드 탐지 프로그램들이 개발되었지만 변종 악성코드를 잡는데 한계가 있음 [4]
- 딥러닝을 이용한 변종 악성코드 탐지 모델을 구현하여, 기존 악성코드 탐지 프로그램과 탐지율을 비교하여 모델의 성능을 검증

딥러닝을 적용한 악성코드 탐지 흐름

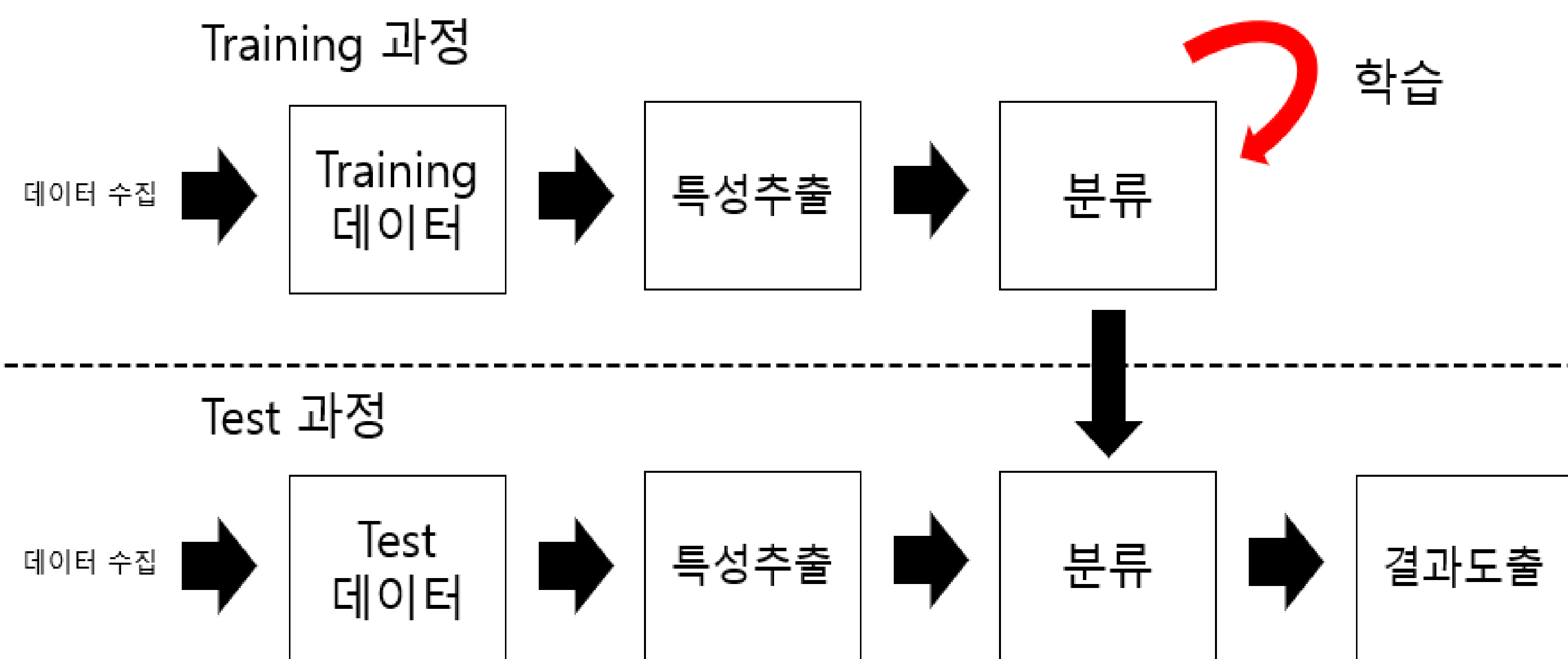


Figure 1. 딥러닝을 적용한 악성코드 탐지 흐름도

- 악성코드 탐지 실험 환경
Training Data : 5종류 악성코드 (A, B, C, D, E)
Test Data : 5종류 변종 악성코드 파일 및 일반파일 (변종A, 변종B, 변종C, 변종D, 변종E, 안전한 파일)
- 변종 악성코드를 탐지하기 위해 Training Data를 학습

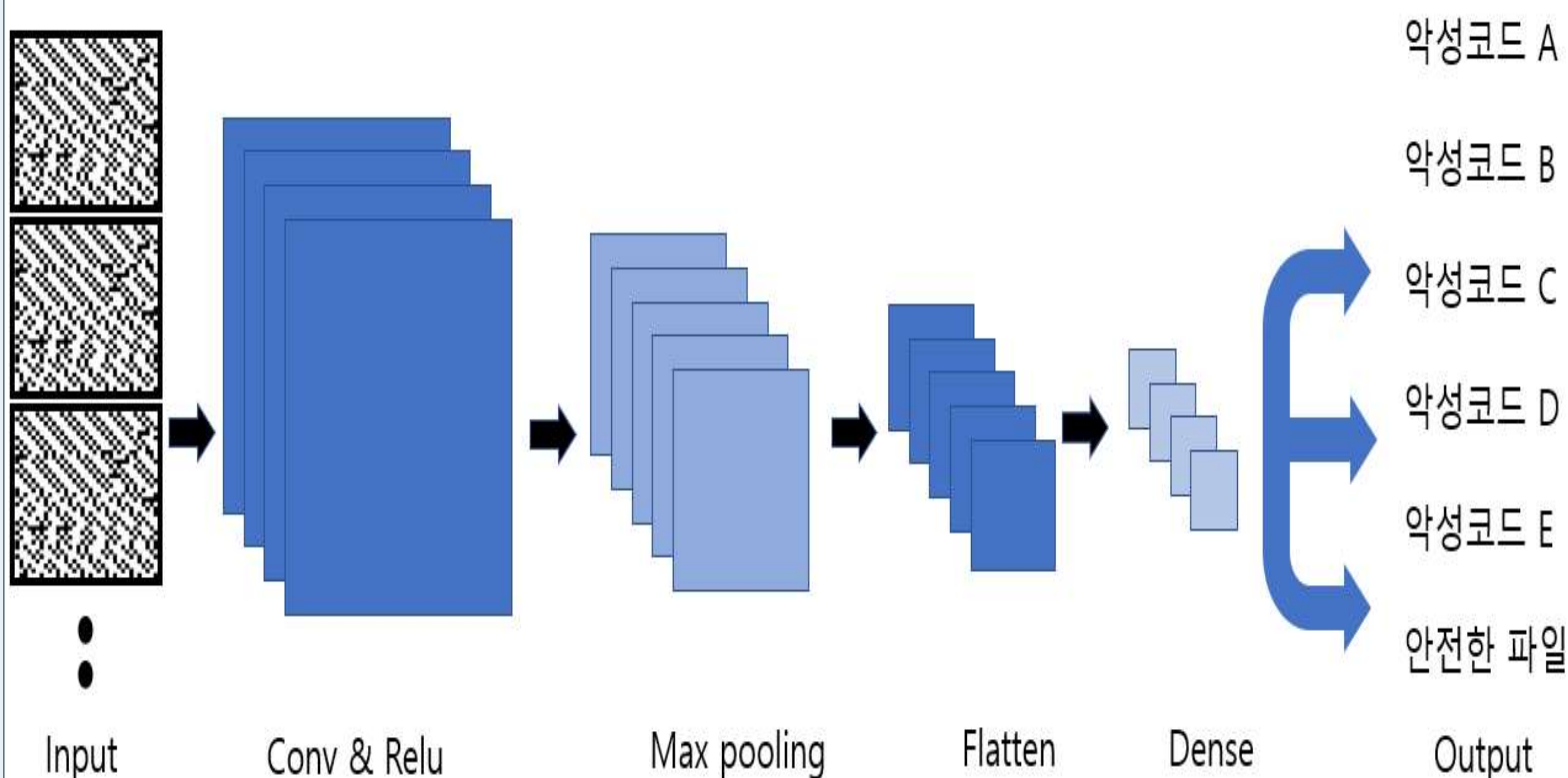


Figure 2. CNN 신경망 구조

- 이미지화 된 Input data는 Relu함수를 통해 2번의 Conv과정을 수행
- Flatten layer 과정을 통한 입력 값들을 벡터로 변환
- Dense layer로 이미지 특징을 추출하여 악성코드 분류

제안한 악성코드 탐지 모델 성능 분석

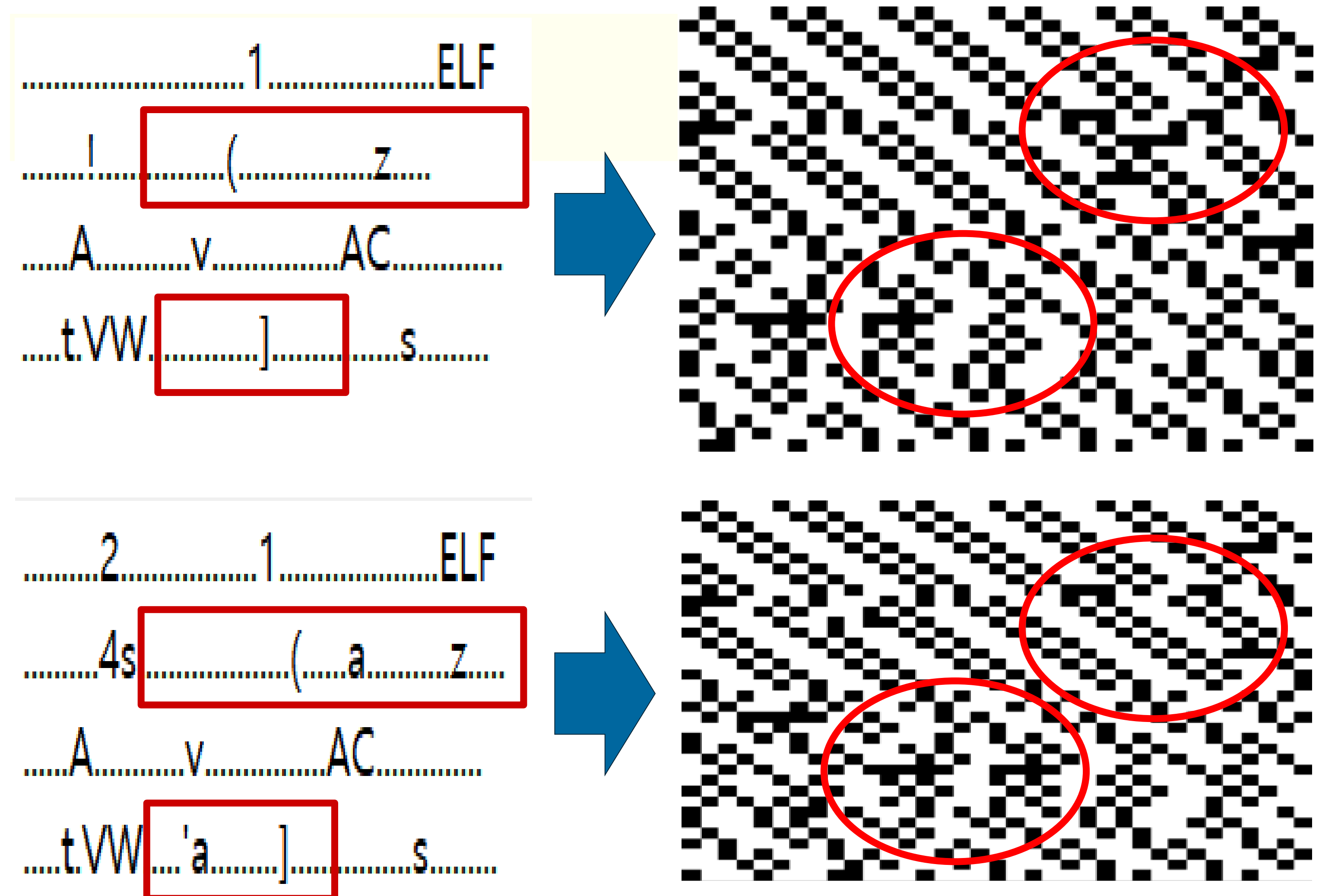


Figure 3. 악성코드D(위)와 변종 악성코드D(아래)의 이미지화 비교

- 악성코드D의 Training Data는 Figure 3. 위의 그림으로 변환
- 변종 악성코드 D의 Test Data는 Figure 3. 아래 그림으로 변환
- 악성코드D와 변종 악성코드D의 시그니처는 유사하기 때문에 비슷한 이미지로 변환
- 변종 악성코드D의 이미지는 악성코드D의 특징을 가지고 있으므로 악성코드D로 탐지

Table 2. 변종 악성코드 탐지 결과

탐지방법	시그니처 기반 탐지	CNN 기반 탐지
A악성코드	3개	10개
B악성코드	4개	10개
C악성코드	3개	9개
D악성코드	4개	9개
E악성코드	1개	10개
안전한 파일	45개	12개
탐지율	30%	98%

- CNN 기반 탐지 모델의 경우 98%의 탐지율을 보였으며, 기존 방식인 시그니처 기반 탐지 모델의 경우 30%의 탐지율을 보였다.

결론

- 본 논문은 딥러닝을 이용한 악성코드 탐지 모델을 소개하고, 변종 악성코드 탐지를 통해 기존 악성코드 탐지 모델과의 성능을 비교함
- 각 악성코드마다 다른 특징을 반영해 실험을 용이하게 진행
- 탐지 결과, 기존 악성코드 탐지 모델과 달리, 딥러닝을 이용한 악성코드 탐지 모델은 변종 악성코드를 대부분 탐지

참고문헌(References)

- [1] 김성기, 장종수, 민병준, "제로데이 공격 대응력 향상을 위한 시그니처 자동 공유 방안," 정보과학회논문지, 제 37권 4호, pp. 255-262, 2010년 8월.
- [2] A. Peris, F. Casacuberta, "NMT-Keras: a Very Flexible Toolkit with a Focus on Interactive NMT and Online Learning," The Prague Bulletin of Mathematical Linguistics, vol. 111, no. 1 pp. 113-124, sep. 2018.
- [3] 문재연, 장영현, "랜섬웨어 분석과 피해 최소화 방안," JCCT, 제 2권 1호, pp. 79-85, 2016년 2월.
- [4] 최선오, 김영수, 김중현, 김익균, "딥러닝을 이용한 악성코드 탐지 연구동향," 정보보호학회지, 제 27권 3호, pp. 20-26, 2017년 6월.

연락처(Contact Information)

아주대학교 우호성, 정건웅, 고준영, 김재현

E-mail : whs1126@ajou.ac.kr, zoavm111@naver.com, kdb2658@ajou.ac.kr, jkim@ajou.ac.kr